Project by: Avani Adhikari

Project Summary

Kathmandu, the capital city of Nepal, has been continuously settled for 2000 years. This long history coupled with unplanned and rapid urbanization in recent years has created a city that experiences significant environmental risks such as flooding, landslides. It is important to understand how these risks are spatially spread out across the city, especially as they relate to the vulnerable population.

However, as Nepal is a developing country, fine-scale data on such matters are difficult to find. In this project, we sought to overcome the data gap by using satellite imagery, open source data, and geospatial AI techniques and create a vulnerability map of Kathmandu.

Building Footprint from Open Street Maps was used to train a Random Forest Model and applied to Landsat 8 imagery to detect built up areas. Similarly, using the GEE library in python, Landsat 8 imagery was used to detect vegetation in the city by calculating the NDVI index.

Flood vulnerability was derived from inundation estimates using the DEM model from NASA. Finally, ward-level household wealth data from the 2021 Nepali census was used as a measure for social vulnerability. After normalizing all of these datasets and assigning them semi-equal weights, the final map was created.

Methodology

The project is available to be viewed on the github repository at https://github.com/avavani/kathmandu_liveability. Overall, 4 jupyter notebooks take us through the process.

1_building_footprint.ipynb and 3_land_classification.ipynb help us download and classify our building footprint.

2_flooding.ipynb models inundation whereas 4_vulnerability.ipynb aligns, normalizes, and combines these various dataset.

As wealth distribution data is only available in tabular form from the census, a separate process was used to spatialize it. The methods used for deriving the wealth estimates can be found here: https://avavani.github.io/kathmandu_urbandata/docs/index.html

Building Footprint Classification

In notebooks 1 and 3, we download training data and classify building footprint using a random forest model.

Raw data for building footprint was taken from Open Street Maps. First, the Overpass API was used to query for buildings within the boundary of Kathmandu. The JSON response from the API was converted into GeoJSON using a recursive for loop. The conversion process involved extracting three main elements from the OSM data: nodes, ways, and relations.

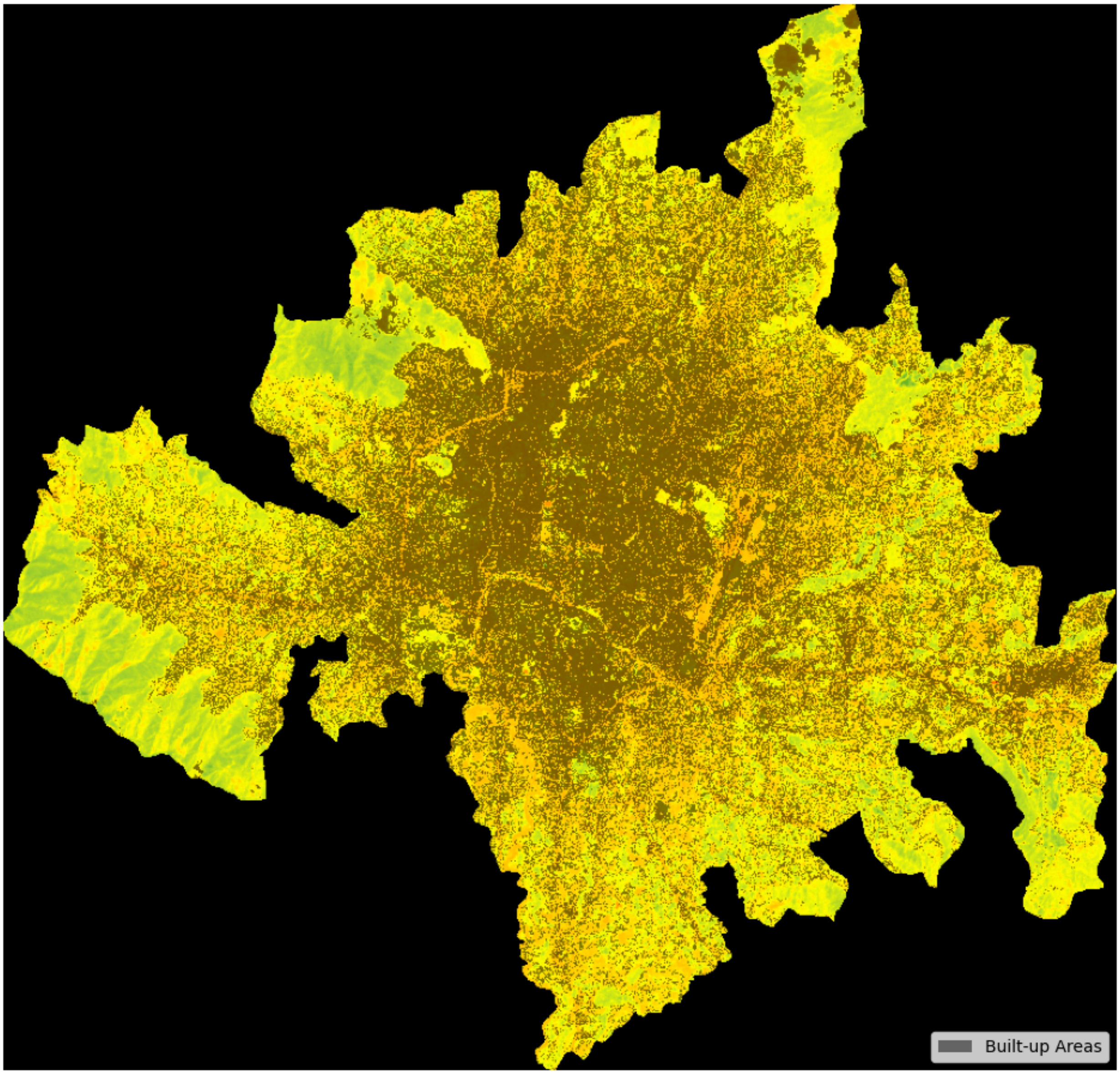
Nodes containing geographical coordinates were stored in a dictionary for quick lookup. Ways representing building structures were processed by collecting their node coordinates and converting them into closed polygons, ensuring they had at least four points to create valid polygon geometries. Each building's tags were preserved as properties in the GeoJSON feature, to be used for future refinement. Finally, the resulting GeoJSON collection was organized into a feature collection structure and saved.

We also needed to download landsat imagery for classification. First, we wrote a function to convert geodataframe objects into google earth engine geometry. This is done for future calculations so that we can use the bounds of Kathmandu for earth engine processes. We then wrote two functions that get and download a yearly snapshot of Kathmandu from the Landsat database. Landsat 7 is used if the date is before 2013 and Landsat 8 for after. Images were then filtered by cloud cover and the least cloudy image was selected. In order to make NDVI calculation easier, we also used Google Earth's built in function to calculate and download this image remotely. The final output was the required Landsat imagery, as well as NDVI files.

The vector building footprint was rasterized and reshaped to fit the bounds of landsat. The dataset was divided into training and testing sets using a 70/30 split ratio, and stratification was applied to maintain the same proportion of built-up areas in both sets. A Random Forest Classifier was implemented with 100 decision trees, a maximum depth of 15. The model was trained using all available processor cores to maximize computational efficiency.

After training, both training and testing accuracies were evaluated to assess model performance. The testing was done for 2024 Landsat data and gave us an accuracy of 69%. In order to ensure that we received the most up to date information on built area, the trained model was applied to new Landsat Imagery from 2025. The final prediction was saved as a GeoTIFF.

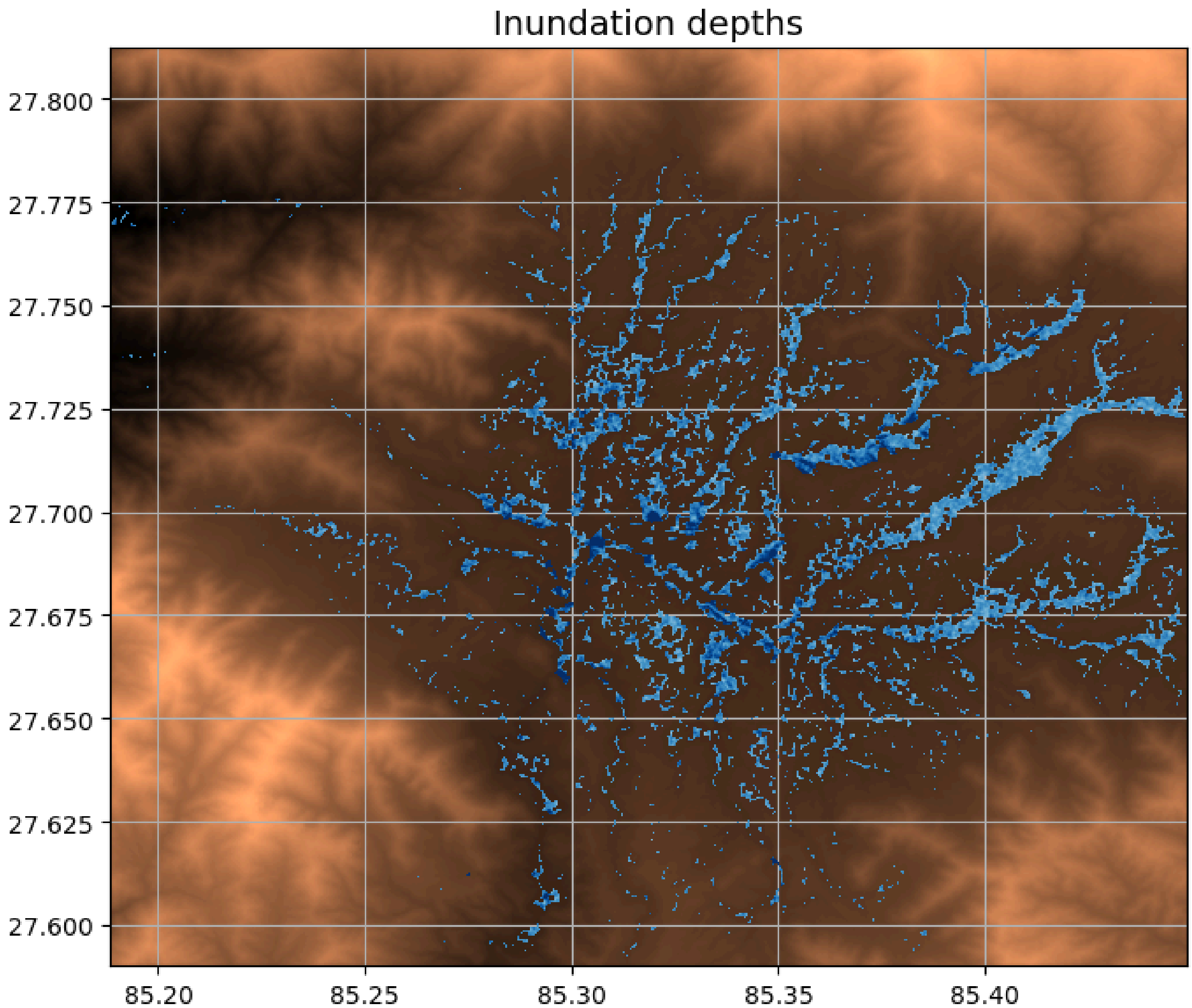
Kathmandu 2025: NDVI with Built-up Area Overlay



Inundation

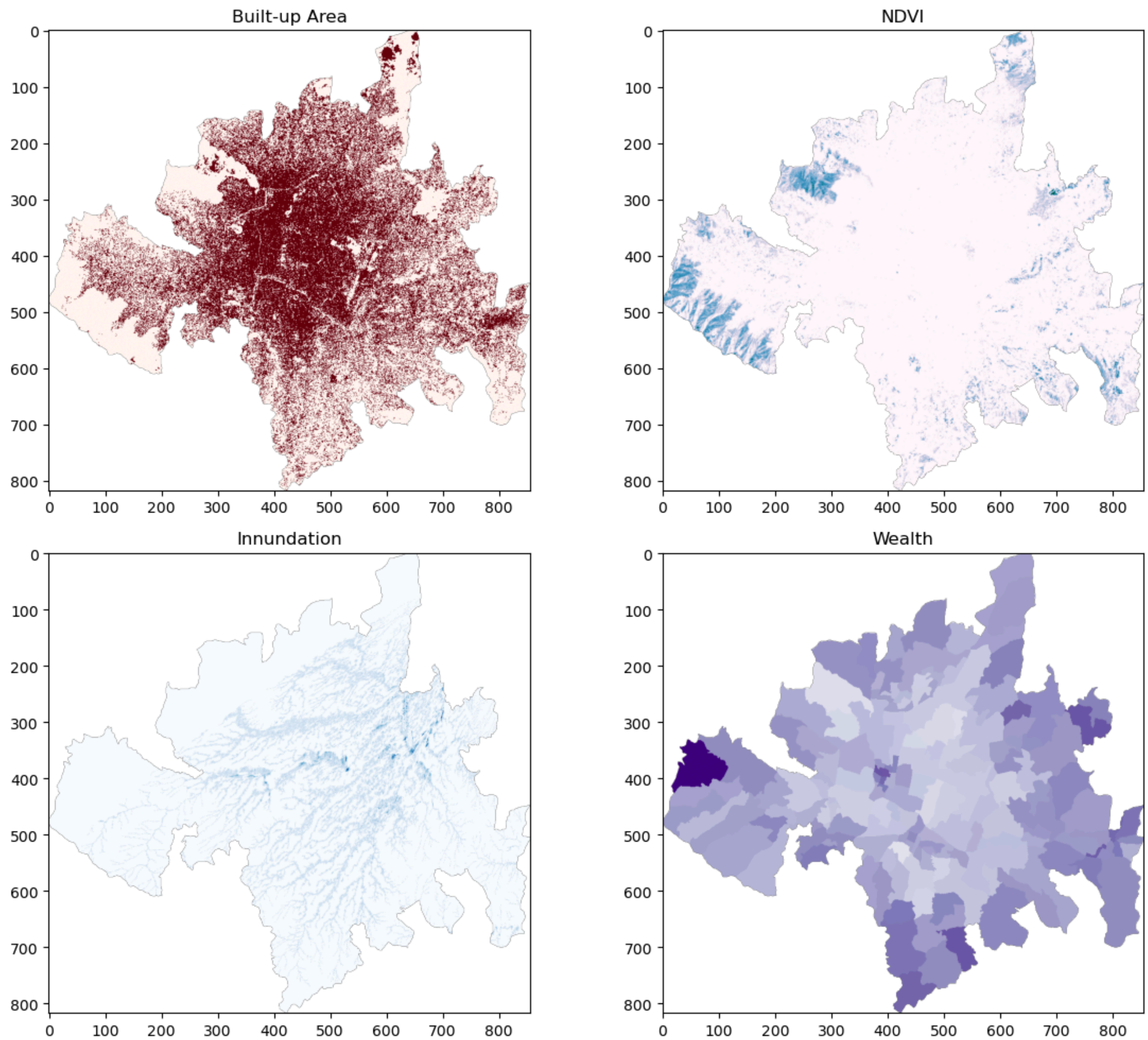
In notebook 2, we download DEM data from NASA's NASADEM_HGT dataset for the tiles covering Kathmandu. The raw data was in HGT format so the GDAL translation tool was used through a subprocess call in order to convert it into GeoTIFF. The DEM was then clipped to match the study area and enhanced by applying a 2x scale factor with cubic resampling.

This enhanced DEM was then processed using the PySheds library for hydrological modeling. The workflow included conditioning the DEM by filling pits and depressions and resolving flat areas. D8 flow direction was calculated to understand where water would flow, followed by flow accumulation to identify stream networks. HAND was computed to understand terrain characteristics relative to drainage networks. The resulting inundation extent was exported as a GeoTIFF.



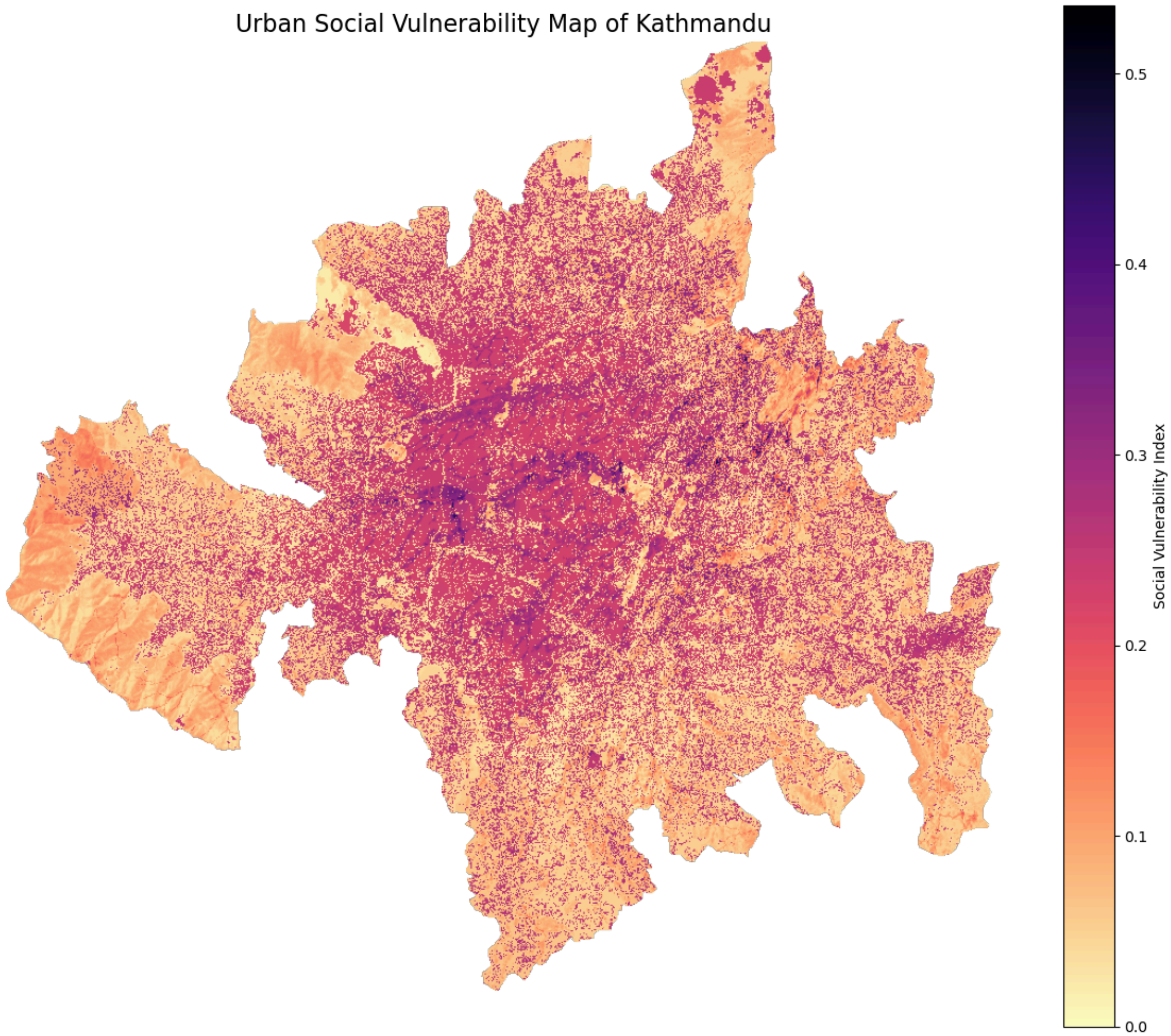
Vulnerability Mapping

All of these datasets were then combined in notebook 4 to create the final vulnerability map. First, the various datasets were loaded, aligned, and clipped to ensure that all of them share the same geographic extent and coordination system. We also mask the datasets to the city boundary.



Wealth data was called and transformed. The raw data divides the population in a ward into 5 economic quartiles ranging from poorest to richest. We took the percentage of population in each quartile and multiplied it with weighted values to create an index where higher weights are assigned to poorer households. Now, poorest households contribute fully to vulnerability while richest contribute 0. Finally, vector wealth data was raterised into a continuous surface to match our other datasets.

All input variables were then normalized, and NDVI values were inversed as lower vegetation indicates higher risk to landslides. Negative and NaN values were replaced with appropriate defaults. Finally, a weighted overlay approach was implemented to calculate the total vulnerability score, with weights of 0.2 for built-up density and NDVI, and 0.3 for flood inundation and wealth vulnerability.



Findings

Plotting the vulnerability index over folium basemap, we were able to validate some areas of extreme vulnerability with ground truth data. Clusters of settlement in the low-income regions of central Kathmandu and flood-prone regions of suburban Kathmandu were shown to have the highest vulnerability score. As such, we accept that the vulnerability map displays to some accuracy the ground reality.

In the future, we would like to supplement our map with more social indicators such as housing quality for a comprehensive understanding of vulnerability.